*Supplement of*

# Computation of backwater effects in surface waters of tidal lowland catchments including control structures – An efficient and re-usable method implemented in the hydrological open source model Kalypso-NA (4.0)

Sandra Hellmers[1], Peter Fröhle[1]

[1]Institute of River and Coastal Engineering, Hamburg University of Technology, Hamburg, 21073, Germany

*Correspondence to*: Sandra Hellmers (s.hellmers@tuhh.de)

## Content

### S1.   The open source model KalypsoNA (4.0) and Kalypso Hydrology

The hydrological open source model Kalypso-NA is a well-applied semi-distributed, deterministic, multi-layered and combined conceptual-physical based model. The source code of the model Kalypso-NA is based on the model BCENA, first developed by Björnsen Consulting Engineers (BCE) and renamed as well as given over to the institute of river and coastal engineering of the Hamburg University of Technology (TUHH) in the years around 2000. The continuous development over more than 20 years is driven by research demands and serves for application purposes in practice.

The model supports the simulation of hydrological processes on the meso scale in the soil, vegetation, atmosphere and water compartments: such as surface runoff, infiltration, snow, evapotranspiration, soil moisture, interflow, base flow and groundwater recharge (Fig. S1). The meso scale modelling is enhanced recently to model local scale drainage measures (LSDMs). These measures are also known as sustainable urban drainage measures (SUDS) , green infrastructure (GI) or best management practice (BMP) (Hellmers and Fröhle, 2017).

35    The model is well applied for (1) analysing the actual state of a catchment, (2) efficient flood forecasting in operational models and (3) modelling flooding under changed conditions caused by climate change, urbanisation, mean sea level rise and adaptation measures.

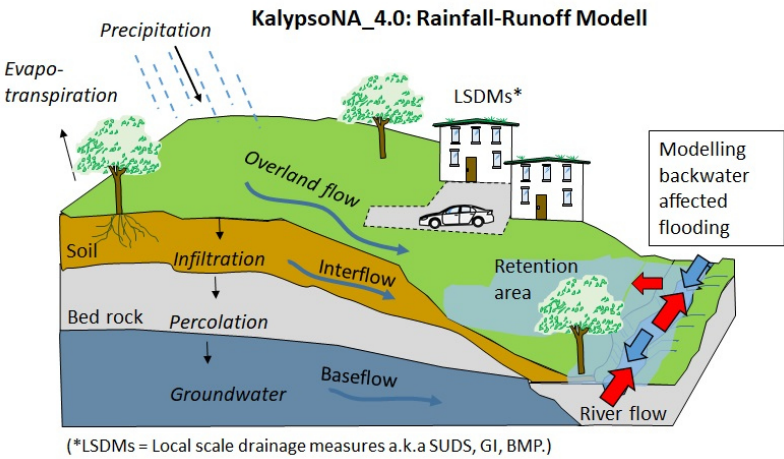



**Figure S1: Modelling hydrological processes using the rainfall-runoff model Kalypso_4.0 including the features to model backwater**
40    **effects in streams and adjacent (retention) areas.**

The model Kalypso-NA is a compiled Open Source calculation code written in FORTRAN under the license LGPL (available under http://kalypso.wb.tu-harburg.de/downloads/Kalypso-NA/). It is integrated in the existing and in practice well approved software module Kalypso Hydrology (Belger et al., 2009; Lippert et al., 2009; Hellmers et al., 2016). Among others, this module provides a graphical user interface with a workflow view. It is part of the open source project Kalypso (available under:

45    https://sourceforge.net/projects/kalypso/), which is an application for geospatial and hydrological as well as hydrodynamic-numerical modelling. The outline of the Kalypso project is illustrated in Fig. SS2 including details of the hydrological module. Further modules are Kalypso Hydrodynamic (1D2D & WSPM), Kalypso Flood and Kalypso Risk. Additional non-public modules include Kalypso Evacuation (Lippert et al., 2009) and the "Prognosesystem" Kalypso (Schröder and Lippert, 2006). The pre-processing of input parameters for the execution of the calculation code is done with ASCII files.

50

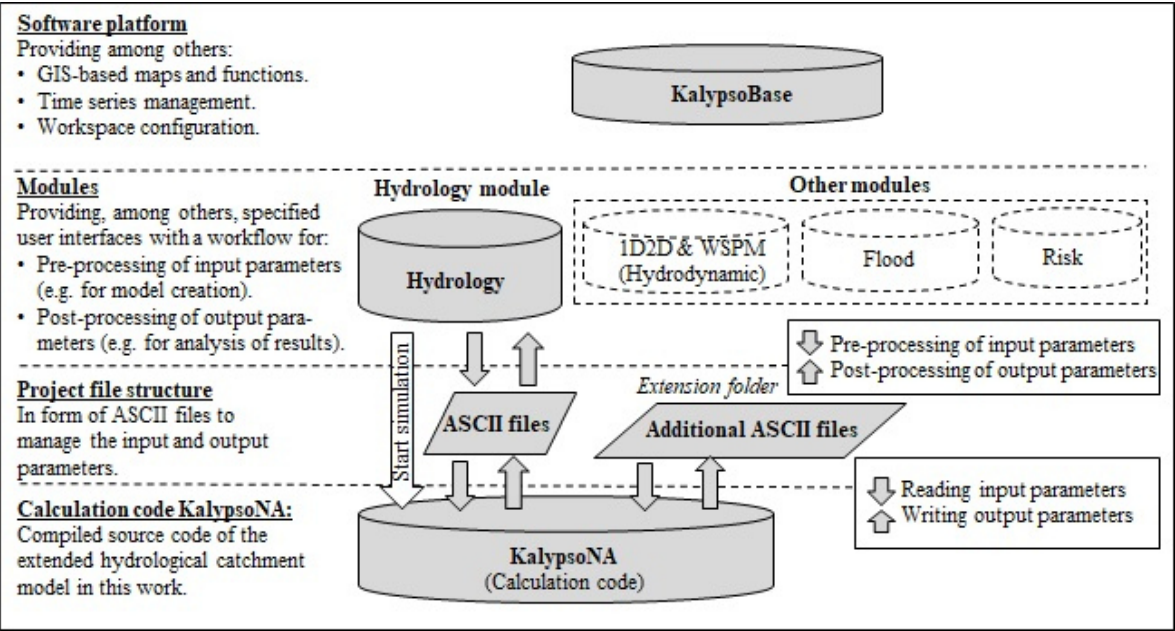



**Figure S2: Outline of the Kalypso project with an overview of public modules and details of the module Kalypso Hydrology as well as the extended calculation code (Kalypso-NA). Input and output parameters for backwater effect computation are processed in ASCII files within an extension folder (Hellmers, 2020).**

55

**S2. Input parameters to compute the flood routing and backwater effects in streams and adjusted lowland areas**

| ASCII file | Description of input parameters |
|---|---|
| strands.ctrl | Configuration of criteria for control structures of linear data structures (streams). |
| strands.kami | Input parameters for the flood routing computation in stream segments on meso scale. |
| strands.topo | Input parameters to describe stream profiles. |
| nodes.xyz | Geographical location defined by the coordinates x (longitude), y (latitude) and z (height) for junction nodes. |
| "node-name".w | Time series of water levels (e.g. of gauging stations) for defined junction nodes. |
| "strand-name".wvq | (Optional) input parameters of the WVQ-relations for example from existing hydrodynamic model results when using the KM5-method. |

**strands.ctrl:**

```
Input parameters and explanation:

m_ctrl_flag                        Kind of driver for the control function
m_ctrl_element                     Name of element to which the threshold value is dedicated
m_ctrl_value                       Threshold value to start the control function
m_ctrl_duration                    Minimum duration of the control function after start
m_ctrl_value_end                   Threshold value to end the control function
m_ctrl_duration_end                Additional duration of the control function to end
m_ctrl_wvqFunction                 File of control function parameters
m_flag_external_timeseries         Query if the time series is computed during the run-time
                                   (=false) or given as external (pre-set) time series (=true).
m_advance_ctrl_duration            Forecast function: Start of control function before
                                   threshold value is reached
m_advance_ctrl_target_value        Forecast function: control function activation till a
                                   target volume in the control element is reached
m_mother_backwaterSystem           (optional) Order number of the interactive backwater system
m_min_activation_duration          Minimum duration of the control function

Example of a control structure (distinct name: 6031ctrl) with five control functions:

6031ctrl 5
waterlevel 3160 -9999.0 120 -9999.0 120 6031_0.wvq false 0 0 0 0
waterlevel 3160 -0.95 30 -0.98 30 6031_01.wvq false 0 0 0 0
waterlevel 3160 -0.85 30 -0.87 30 6031_1.wvq false 0 0 0 0
waterlevel 3160 -0.80 30 -0.83 30 6031_2.wvq false 0 0 0 0
waterlevel 3160 -0.75 30 -0.77 30 6031_3.wvq false 0 0 0 0
ENDE
```

60  **strands.kami:**

```
Input parameters and explanation:

this_strand_name_gml          Distinct name of river stream section
m_bed_width_m                 (optional angular profile) Profile bed width (m)
m_bank_gradient               (optional angular profile) Bank gradient (m),
m_roughness_kst_ms            (optional angular profile) Manning-Strickler roughness kst (m1/3/s),
m_bankfull_height_m           (optional angular profile) Bankfull height (m),
m_hydr_diameter_m             (optional circular profile) Hydraulic diameter (m)
m_roughness_ks_m              (optional circular profile) Equivalent sand roughness ks (m),
m_max_steps                   Number of supporting points to compute the WVQ-functions
m_crossSection_form           Cross section form (circular or rectangular),
m_calc_method                 Choice of the computational approach (Manning-Strickler or Darcy Weisbach).

Example of three strands with rectangular profile; computation method: manning-strickler:

3160km1 7.0 1.0 30 3.3 0 0 25 rectangular manning
3161km1 13.0 1.5 30 4.3 0 0 25 rectangular manning
3170km1 25.0 1.0 30 2.3 0 0 25 rectangular manning
```

**strands.topo:**

```
Input parameters and explanation:

this_strand_name_gml          Distinct name of river stream section
m_strand_length               Stream length (m) (optional if geographical data not given or an exact flow
                              path length is defined)
m_strand_gradient             Gradient (-) (optional if geographical data not given)
m_strand_bedLevel_mNN         Lowest bed level (m a.s.l),
m_min_waterLevel_mNN          Minimum water level (m a.s.l.)
m_strand_wq_flag              Flag for: given w-q data = 1, w-q curve = 2, computet kalinin curve = 3 or
                              ctrl storage strand = 4
m_factor_Q_WQ_relations, m_adding_W_WQ_relations, m_adding_A_WQ_relations: (optional adjustment factors)
                              discharge (default = 1.0), water level variance (default = 0.0 m), cross-
                              section adjustment (default = 0.0 m²).
m_strand_reverseSystemNumber  Order number of the backwater system
m_min_backwater_h_m           Minimum tolerable backwater level difference (m)
m_strand_calc_kalinin_flag    Flag for KM computation method = 1; KM parameters pre-set = 0; Flag = 4 if
                              it is a ctrl-element.
m_nrOfCatchments                        (optional) number of adjusted spatial data structures
strand_overflow_overlay_id(iCatchm)     (optional) linked spatial structure
m_overflow_height_mNHN(iCatchm)         (optiona) overflow height in m a.s.l.

Example of five strands and one with an adjusted lowland area:

3151km1 1397.0 0.001 -1.60 -0.87 3 0.0 0.73 0 6 0.00001 1 1
24 -0.4
3152km1 493.0 0.001 -1.92 -0.88 3 0.0 1.04 0 6 0.00001 1 0
3160km1 1026.0 0.005 -2.30 -0.89 3 0.0 1.41 0 6 0.00001 1 0
3161km1 4685.0 0.005 -2.35 -0.91 3 0.0 1.44 0 6 0.00001 1 0
3170km1 247.0 0.002 -2.35 -0.92 3 0 1.43 0 6 0.00001 1 1
```

**nodes.xyz**

```
Input parameters and explanation:

m_node_id          Distinct name of junction node in the hydrological network
m_descr_node       (optional) description
m_x_coordinate     X-coordinate
m_y_coordinate     Y-coordinate
m_z_coordinate     Z-coordinate
m_factor_flowpath  (optional) Prolongation of computed flow path length to the linked element in the
                   hydrological network

Example:

3151 Moorfleet_3 572087.7859 5929285.224 -0.99 1.0
3161 Moorfleet_2 572651.6114 5928117.698 -0.99 1.0
3140 Moorfleet 4 572414.6276 5929117.823 -0.99 1.0
```

65

**"node-name".w**

```
Input parameters and explanation:

"node-name".       Distinct name of junction node in the hydrological network
.w                 Time series of "w" = water level; alternatives: p = precipitation and q = discharge
201101010000       Start date: yyyymmddmmss
Grap               Format

Example:

      201101010000
grap
01.01.2011 00:00:00     2.000
01.01.2011 00:05:00     2.030
01.01.2011 00:10:00     2.050
01.01.2011 00:15:00     2.080
01.01.2011 00:20:00     2.100
01.01.2011 00:25:00     2.120
01.01.2011 00:30:00     2.140
…
…
```

**"strand-name".wvq**

```
Input parameters and explanation:

" strand-name".          Distinct name of stream section with control function in the hydrological network
.wvq                     pre-set water level-volume-discharge function

Example:

SPEICHER    1155    6031_ov    6031    6031
Fakt_SeeV 1.00
text;text
6031ctrl    0.000135  0.009000  0.000000 201
      -1.05         0.000045         0.30         0.000         0.000
      -1.00         0.000090         0.30         0.000         0.000
…
…
```

70 **Explanation:**

The input parameters, to model the flood routing with the KM-method, comprise the profile data of stream segments on different scales. For the geometrical profiles using the KM1-method, the parameters are given in the file "strands.kami". In this file the parameter values of district and meso scale streams are written. The file comprises the following data: profile bed width (m), bank gradient (m), Manning-Strickler roughness $k_{st}$ (m$^{1/3}$/s), bankfull height (m), hydraulic diameter (m), equivalent sand

75 roughness $k_s$ (m), number of interpolation steps (-), cross section form (circular or rectangular), minimal longitudinal gradient (m/m) and a choice of the computational approach (Manning-Strickler or Darcy Weisbach). According the chosen approach (Manning-Strickler or Darcy-Weisbach) and the cross section (circular or rectangular) the mandatory input parameters are specified. The topographical data of stream segments and the input parameters for the backwater effect computation are given in the file "strands.topo". The stream length (m) and gradient (-) are computed with the geographical data of the upstream and

80 downstream junction nodes or can be provided as user defined input values. A flow path prolongation factor is given optionally per data structure in the hydrological network. The stream segment input parameters comprise: the lowest bed level (m a.s.l), the minimal water level (m a.s.l.) and the number of supporting points to compute the WVQ-functions.

For calibration purposes an adjustment of the WVQ-functions per stream segment profile is provided (optionally) in the input files in the form of a percentage factor for the discharge (default = 1.0), a water level variance value (default = 0.0 m) and a

85 cross-section adjustment using a variance value (default = 0.0 m$^2$). The stream segments are part of different backwater systems which are defined with an ordinal number (1 to n) from downstream to upstream. The tolerable backwater level difference is defined in (m) per stream segment, whereas larger values are suggested for streams with wider profiles than for narrow ones. Another input parameter in the file (here: "strands.topo") gives the number of linked spatial data structures (for example, riparian or retention areas). The indexes of the linked spatial data structures are listed in the input file with an overflow height

90 in m a.s.l. When the water level of a stream segment reaches this overflow height, water is flowing into the free storage volume of the linked spatial structure.

**S3.  Hydrological network creation**

In hydrological numerical models, the network represents the interlinked order among three main types of elements: (1) directed stream segments (namely river segments, reservoirs, pipes, ditches or open rills), (2) junction nodes and (3) spatial

95 structures such as subcatchments. The processes in stream segments are computed with flood routing methods. Each stream segment is connected with an inflow and outflow junction node. The junction nodes function as joint connections to set rules of flow redistribution in the network interconnections. Subcatchments configure the spatial and temporal parameters of drained

areal compartments within the network. Any spatial data structure in the network has to be defined with an explicit position by the order of stream segments and the respective outlet junction node.

100    The applied method in this work is based on the theory of the Shreve's stream order (Shreve, 1967). The computational order is based on a directed data tree structure with an explicit start and an explicit end according to the stream segments along the main stream on the meso scale. It defines a directed graph with incoming tributaries. The developed algorithm in the applied model Kalypso Hydrology prevents the generation of closed loops in circular order among any network element (Hellmers et al., 2016).

105    **S4.    Description, equations and code sections of the conceptual flood routing approach**

### S4.1    Description of the flood routing approach of Kalinin and Miljukov

The approach of (Kalinin and Miljukov, 1957) divides a stream into a number of characteristic lengths. Each length is considered to be small enough for assuming a quasi-stationary relationship on the basis of a hysteresis curve (see Fig. S3).

110    **Figure S3: (a) Parametrisation of a characteristic length Lc of a stream segment. (b) Illustration of a hysteresis curve, which describes the relation between discharge and water level for instationary and stationary conditions. Similar illustrations are given in (Hellmers, 2020; Maniak, 2016; Koussis, 2009).**

The Kalinin and Miljukov (KM) approach is based on the assumption that in the quasi-stationary state, the discharge Q (m³/s) is directly related to the water volume V (m³) per characteristic stream segment. The slope of the function V = f(Q) results in

115    the retention coefficient K (s). Therewith, the relation $\Delta V = K * \Delta Q$ is derived for each characteristic length $L_c$ of a stream segment. The conditions under the instationary water level gradient $I_w$ are compared with the stationary water level gradient $I_{stat}$. For each point in the discharge hydrograph (for example, $Q_1$), three corresponding water level stages are given in the hysteresis curve (Fig. S3 (b)). The water levels at the rising and falling limp of a stationary discharge curve appear a time step size Δt later than in the instationary flow process. The water-level-discharge-relation (WQ-relation) is derived by assigning

120    the discharge $Q_{t-1}$ to the water level W(t). For a characteristic length $L_c$ the water levels in a stationary and instationary conditions are labelled in Fig. S3 (a). The average water level at point (m) is transferred to point (r) on the basis of the WQ-relation. The volume in the stream segment in the instationary state (indicated with the points: A,B,G,F) corresponds to the stationary state (indicated with: C,D,G,F). This results in a distinct water-level-volume-discharge-relation (namely "WVQ-relation"), which is valid for a characteristic length $L_c$ with a constant water level slope $I_w$ and an even cross-section as well

125    as roughness along the stream length. This approach results in the computation of the instationary flow of a complete reach by a cascade of characteristic stationary linear reservoirs. Using the Nash-model principle with the gamma-function according to Dooge [1973] it becomes more flexible in using a real number of characteristic lengths. The parametrisation to apply this approach depends on profile geometry, slope and roughness coefficients. An analysis by Dooge [1973] shows that the KM-approach can closely reproduce the results of the linearized Saint-Venant solution.

**S4.2 FORTRAN code: computation of the flood routing parameters**

```fortran
subroutine calc_kami_strand (iStrand_ordinal)
    !input
    integer, intent (in)                  :: iStrand_ordinal

    !locals
    type (Strand), pointer                :: m_act_strand
    type (strand_KM_calc), pointer        :: m_strand_KM_calc
    type (strand_km_input), pointer       :: m_strand_km_input
    type (strand_km_results), pointer     :: m_strand_km_results
    integer                               :: max_steps, step_i, size_wvq
    real(dp), allocatable                 :: volume_V (:), discharge_Q (:), discharge_Q_m (:),
                                               discharge_Q_delta (:), length_Lc_i (:)! results
    real(dp)                              :: length, strand_gradient ! input data
    real(dp)                              :: slope_result_VQ
    real(dp)                              :: length_Lc_circularProfile,
                                               Q_full_circularProfile

    !preparation of KaMi-Routing computation
    m_act_strand => strands(iStrand_ordinal)
    m_strand_KM_calc => m_act_strand%m_strand_KM_calc
    m_strand_km_input => m_strand_KM_calc%m_strand_km_input
    m_strand_km_results => m_strand_KM_calc%m_strand_km_results

    max_steps = m_strand_km_input%max_steps !

    !**************************************************

    strand_gradient = m_act_strand%strand_gradient
    length = m_act_strand%strand_length

        if (m_strand_km_input%crossSection_form == 'circular') then
        !for pipe flow only the empty and full stage are considered
        size_wvq = 2
        else
        size_wvq = max_steps+1
        end if

    allocate (volume_V (size_wvq))
    allocate (discharge_Q (size_wvq))
    allocate (discharge_Q_m (size_wvq))
    allocate (discharge_Q_delta (size_wvq))
    allocate (length_Lc_i (size_wvq))
    allocate (m_strand_km_results%discharge_Q_m3s(size_wvq))
    allocate (m_strand_km_results%waterlevel_h_mNN(size_wvq))
    allocate (m_strand_km_results%crossSection_A_m2(size_wvq))
        ! the computation of the Kalinin-Miljukov Parameters is defined in a subroutine which can be used
        by other routing elements
        call calc_kami_parameters (m_act_strand%gml_name, size_wvq,
                m_strand_km_input%bankfull_height_m, m_strand_km_input%bed_width_m ,
                m_strand_km_input%bank_gradient, &
                m_strand_km_input%roughness_kst_ms, length, strand_gradient, &
                m_strand_km_input%hydr_diameter_m, m_strand_km_input%roughness_ks_m, &
                m_strand_km_input%crossSection_form, m_strand_km_input%calc_method, &
                crossSection_A_m2)                                      (… next page)
```

```fortran
                    !results:
                    volume_V, discharge_Q, discharge_Q_m, &
                    discharge_Q_delta, length_Lc_i, m_strand_km_results%waterlevel_h_mNN,
                    m_strand_km_results%discharge_Q_m3s, m_strand_km_results%crossSection_A_m2)

        m_strand_km_results%km_l_length = sum(length_Lc_i)/(size_wvq-1)
        m_strand_km_results%km_n_storages = nint(length / m_strand_km_results%km_l_length)

    !implementation of a slope function for angular profiles.
        (Y_values; X_values)
    if (m_strand_km_input%crossSection_form == 'circular') then
            !computation according to (Euler, 1983)
                m_strand_km_results%km_k_retConstant_h = 0.64_dp *
                m_strand_km_results%km_l_length *
                (m_strand_km_input%hydr_diameter_m**2)/m_strand_km_results%discharge_Q_m3s(2) /3600.0_dp
                ![h]
    else
        call calc_slope_function (size_wvq, volume_V, discharge_Q, slope_result_VQ) !input
                is here the discharge and volume to compute the slope function

         m_strand_km_results%km_k_retConstant_h = (slope_result_VQ /
                m_strand_km_results%km_n_storages)/60/60 ! transfer from [s] to [h]
    end if
        !Setting the results of the Kalinin-Miljukov method to the strands data for the
        backwater routing:
    call allocate_strands_wqa (size_wvq,iStrand_ordinal)

    do step_i = 1, size_wvq
        m_act_strand%strand_waterLevel (step_i) = m_act_strand%strand_bedLevel_mNN +
        (m_strand_km_results%waterlevel_h_mNN(step_i)+m_act_strand%adding_W_wq_relations)
        m_act_strand%strand_discharge (step_i) =
        m_strand_km_results%discharge_Q_m3s(step_i)* m_act_strand%factor_Q_wq_relations
        m_act_strand%strand_crossSection (step_i) = (m_strand_km_results%crossSection_A_m2
        (step_i)+m_act_strand%adding_A_wq_relations)
        m_act_strand%strand_storageVolume (step_i) =
        (m_strand_km_results%crossSection_A_m2 (step_i)+m_act_strand%adding_A_wq_relations)
                * length
    end do
    m_act_strand%max_wqa_size = size_wvq

end subroutine calc_kami_strand
```

```fortran
subroutine calc_kami_parameters (m_gml_name, size_wvq, bankfull_height_m, width, &
        bank_gradient, kst, length, strand_gradient, &
        m_hydr_diameter_m, m_roughness_ks_m, m_crossSection_form, m_calc_method, &
        !results:
        volume_V, discharge_Q, discharge_Q_m, discharge_Q_delta, length_Lc_i, &
        waterLevel_mNN, discharge_Q_m3s, crossSection_A_m2)
        !input
        character (len = maxLineLength), intent(in) :: m_gml_name
        integer,intent (in)                         :: size_wvq
        real(dp),intent (in)                        :: bankfull_height_m, width, &
                                        bank_gradient, kst, length, strand_gradient
        real(dp), intent(in)                        :: m_hydr_diameter_m, m_roughness_ks_m
        character (len = 15), intent(in)            :: m_crossSection_form, m_calc_method

        real(dp),intent (out)                       :: volume_V (size_wvq), discharge_Q &
                                (size_wvq), discharge_Q_m (size_wvq), discharge_Q_delta (size_wvq), &
                                length_Lc_i (size_wvq), waterLevel_mNN (size_wvq), &
                                discharge_Q_m3s(size_wvq), crossSection_A_m2(size_wvq)
                                ! results which are allocated before
        !locals:
        real(dp)                                    :: h_delta
        integer                                     :: step_i
        real(dp)                                    :: h_act_i
        real(dp)                                    :: velocity_act_i, velocity_next_i, &
                                                A_next_i, A_act_i
        real(dp)                                    :: P_act_i, P_next_i, R_hyd_next_i, &
                                                R_hyd_act_i
        !darcy parameter:
          real(dp)                                  :: lambda_act, lambda_next, &
                                        f_shape_coefficient_act, f_shape_coefficient_next
        real(dp)                                    :: length_Lc_circularProfile, &
                                        Q_full_circularProfile
        !result file
        integer                                     :: kamiStreamResultsID
        character(len=81)                           :: kamiStreamResultsFile

        h_delta = bankfull_height_m / (size_wvq-1) ! Computation of the step size for wvq-function
        h_act_i = - h_delta
if (m_crossSection_form == 'circular') then
            !the equation is according to (Euler, 1983):
            !the characteristical length is not changed for full or empty pipe
            length_Lc_circularProfile = 0.4 * m_hydr_diameter_m /(strand_gradient)
             !calculation of the maximal capacity of the drainage pipe according to darcy
                 with the pipe roughness
                A_act_i = pi / 4 * m_hydr_diameter_m**2
                P_act_i = pi * m_hydr_diameter_m
                R_hyd_act_i = A_act_i / P_act_i
                lambda_act=(1.0_dp/(-2.0_dp*LOG10(m_roughness_ks_m/ (14.84_dp * &
                        R_hyd_act_i))))**2.0_dp
            !flow velocity calculation with the Darcy Weissbach approach:
            velocity_act_i = SQRT((8.0_dp * gravity * R_hyd_act_i &
                            *strand_gradient)/lambda_act) ![m/s]
            Q_full_circularProfile = velocity_act_i * A_act_i
            ! setting the parameters: wvq = 1 => empty pipe; wvq = 2 => full pipe
            length_Lc_i(1) = 0.0_dp
            length_Lc_i(2) = length_Lc_circularProfile
            waterLevel_mNN(1) = 0.0_dp
            waterLevel_mNN(2) = m_hydr_diameter_m
            discharge_Q_m3s(1) = 0.0_dp
            discharge_Q_m3s(2) = Q_full_circularProfile
            crossSection_A_m2(1) = 0.0_dp
            crossSection_A_m2(2) = A_act_i                              (… next page)
```

```fortran
      else     !for angular profiles the w-v-q-steps are computed
      do step_i = 1, size_wvq
                    h_act_i = (step_i - 1) * h_delta ! the computation starts with a
                    waterlevel of 0 at step = 1
                    waterLevel_mNN (step_i) = h_act_i
                    !  computation of wetted perimeter and cross-section according to
                          m_crossSection_form
                    A_act_i = (width + bank_gradient * h_act_i) * h_act_i
          P_act_i = width + 2.0_dp * h_act_i * sqrt(1.0_dp + bank_gradient**2)
          A_next_i = (width + bank_gradient * (h_act_i + h_delta)) * (h_act_i +
                          h_delta)
          P_next_i = width + 2.0_dp * (h_act_i+ h_delta) * sqrt(1.0_dp +
                          bank_gradient**2)
                    R_hyd_act_i = A_act_i / P_act_i
                    R_hyd_next_i = A_next_i / P_next_i
                    crossSection_A_m2(step_i) = A_act_i
                    !   computation of velocity according to method: darcy or manning
      if (m_calc_method == 'darcy') then ! Darcy Weissbach method
            call calc_shape_coefficient(h_act_i, width,R_hyd_act_i, m_crossSection_form,
            f_shape_coefficient_act) !According to (BWK, 2009) page 24ff
            ! lambda calculation according to Colebrook White (turbulent flow =
            without the Reynolds term) for the first wvq-step, the lampda and the velocity is zero:
            if (step_i == 1) then
                lambda_act = 0.0_dp
                velocity_act_i = 0.0_dp
            else
                lambda_act=(1.0_dp/(-2.0_dp*LOG10(m_roughness_ks_m/
                    (f_shape_coefficient_act * 14.84_dp * R_hyd_act_i))))**2.0_dp
                !flow velocity calculation with the Darcy Weissbach approach:
                velocity_act_i = SQRT((8.0_dp * gravity * R_hyd_act_i
                    *strand_gradient)/lambda_act) ![m/s]
            end if
            call calc_shape_coefficient((h_act_i+ h_delta), width,R_hyd_act_i,
                  m_crossSection_form, f_shape_coefficient_next)
            lambda_next=(1.0_dp/(-2.0_dp*LOG10(m_roughness_ks_m/
                  (f_shape_coefficient_next * 14.84_dp * R_hyd_next_i))))**2.0_dp
            velocity_next_i = SQRT((8.0_dp * gravity * R_hyd_next_i
                  *strand_gradient)/lambda_next) ![m/s]
      else
            !calculation method = "manning-strickler":
            velocity_act_i = kst * ((R_hyd_act_i)**(2.0_dp/3.0_dp)) *
                  (strand_gradient**0.5_dp)
            velocity_next_i = kst * ((R_hyd_next_i)**(2.0_dp/3.0_dp)) *
                  (strand_gradient**0.5_dp)
        end if
            !   computation of discharge and volume
            discharge_Q (step_i) = velocity_act_i * A_act_i
            discharge_Q_m3s(step_i) = discharge_Q (step_i)
            volume_V (step_i) = length * A_act_i
            !   computation of averaged discharges for W-V-Q-relations (for angular
                  profiles!)
```

*(… next page)*

135

```fortran
        if (step_i < size_wvq) then
            discharge_Q (step_i + 1) = velocity_next_i * A_next_i
            discharge_Q_m (step_i) = (discharge_Q (step_i) + discharge_Q(step_i +
                1))/2.0_dp
        else
            !last discharge at maximum water-level
            discharge_Q_m (step_i) = 0.0_dp ! bankfull discharge is reached in the step
                before.
        end if

        if (step_i == 1) then
            discharge_Q_delta (step_i) = discharge_Q_m (step_i) ! no averaging of the
                first discharge at waterlevel = 0
        else if (step_i == size_wvq) then
            discharge_Q_delta (step_i) = 0.0_dp ! bankfull discharge is reached in the step before.
        no additional averaged discharge is taken in the computation
        else
         discharge_Q_delta (step_i) = discharge_Q_m (step_i) - discharge_Q_m (step_i - 1)
        end if

        if (step_i == size_wvq) then
                length_Lc_i (step_i) = 0.0_dp ! bankfull discharge is reached in the step before.
        else
                length_Lc_i (step_i) = h_delta * discharge_Q_m(step_i) /(strand_gradient *
                discharge_Q_delta(step_i))
        end if

    end do !loop over wvq-steps
   end if ! query about circular or angular profile

end subroutine calc_kami_parameters
```

### S4.3    Differentiation between simple geometrical and complex river profiles

An objective is the definition of a parsimonious and applicable parametrisation for regional scale catchment modelling. Therefore, two approaches are included: one parsimonious / simple and a second more complex approach. The first developed method uses parameters of primary geometrical profiles (trapezoidal, rectangular or circular) with a low number of input parameters to model the directed flood routing. Each stream segment is represented as a single reservoir and therefore this method is labelled as "KM1"-flood routing method. If irregular profiles are present in stream segments and detailed profile data is available, a second approach using an extended flood routing method is applicable which includes a different parametrisation of main channel and forelands. It results in a subdivision of each profile of a stream segment into five reservoirs. Therefore, the method is labelled as "KM5" flood routing method. The developed theoretical approach is the same for both concepts, KM1 and KM5.

The computation of the flood routing in stream segments on the meso scale with natural irregular profiles requires a separate parametrisation of main channel and flood plane areas. These profiles are subdivided into five reservoirs. Two reservoirs represent the characteristics in the main channel, one reservoir is situated at the bankfull stage and two reservoirs are defined to model the flood routing on the foreland (see Fig. S4). Because of the subdivision into five reservoirs, the concept is labelled as "KM5"-method in this work. To resolve the calculation routine through five reservoirs a "polynomial function" is required to compute the WVQ-relations and the Kalinin-Miljukov parameters. A partition factor α describes the fraction of flow which is routed through the main channel and the flood plane area. This fraction is α = 1 as long as the water level is below the bankfull height. The fraction is reduced (α < 1) when the water level exceeds the bankfull height and a percentage of water (= 1 − α ) flows over the flood plane area. The flood routing on the flood plane is computed using different roughness ( $R_{FP}$) parameters as in the main channel ($R_{channel}$). Both flood routing methods, the KM1- and the KM5-method, serve as basis to compute the backwater effects in stream segments with the presented method in this article.

**Figure S4 Scheme to separate a natural irregular profile into the flood plane (FP) area and the main channel to compute the flood routing along a stream with the KM5-method (Hellmers, 2020).**

## S5. Initialisation of the WVQ-functions per stream segment

The initialisation of the parameters for the backwater effect computation is illustrated in Fig. S5. For the computation of backwater effects, the formal parameters of each linear and spatial data structure are initialised. This includes an initialisation of the water level, volume and discharge per time step. Discharges are computed with the flood routing approaches described in section S3. The corresponding water levels and retained water volumes are derived from the calculated WVQ-relations per stream segment. When the volume in the control structure is increased ($V(t) > V(t-1)$), afflux is generated and the flag for afflux conditions is set to 'true'. The difference in volume between time steps ($\Delta V(t-1)$) is revised continuously during the following backwater calculation routines (b) and (c) (Fig. 5, main article). When the volume in the control system is decreased ($V(t) < V(t-1)$) or not changed ($V(t) = V(t-1)$) the flag for afflux conditions is set to 'false' and the volume ($\Delta V(t-1)$) is reduced by the proportion of the changed volume $\Delta V$ which has been processed already in the time step before. The upstream-directed backwater routing is computed if the 'afflux-conditions-flag' is set to 'true'. The downstream-directed backwater routing is computed if the 'afflux-conditions-flag' is set to 'false'.

**Figure S5: Algorithm to initialise WVQ-relations in streams, control structures and areas per backwater system. Illustrated details of the first calculation routine (a) in the algorithm in Fig. 5 of the main article.**

## S6. Backwater effect computation in upstream direction

The algorithm to compute upstream directed backwater effects on the water levels and volumes is illustrated in Fig. S6. If the following queries are true, the upstream backwater effect computation is executed. These queries are called at the beginning of the calculation routine (see: 'Are afflux conditions present?') with the following equation:

$$is\ W_i(t) - W_{i-1}(t) > \Delta W_{min}?\ and\ is\ V_i(t) > V_{i,free}(t)? \tag{1}$$

where the water level $W_i(t)$ (m a.s.l.) and volume $V_i(t)$ (m³) are defined by the WVQ-relation per stream segment with the index i. $\Delta W_{min}(t)$ is the tolerable backwater affected water level rise given for the stream segments (m) in the backwater system. $V_{i,free}(t)$ is the water volume in the segment without backwater effects, which is computed with the flood routing method.

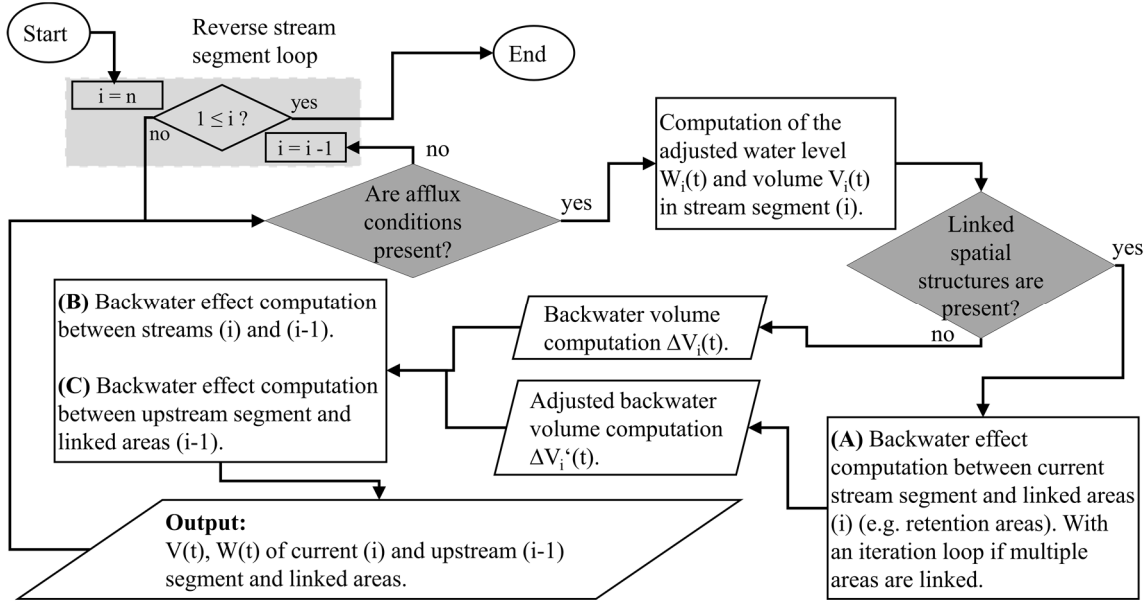



**Figure S6: Computation of the upstream directed backwater effect over stream segments and adjacent lowland areas (e.g. retention areas). Illustrated details of the second calculation routine (b) of the algorithm in Fig. 5 of the main article.**

While afflux conditions are present, the water level in the current stream segment (i) is reduced by the minimum water level difference $\Delta W_{min}(t)$. The adjusted storage volume of the stream segment $V'_i(t)$ is defined accordingly by the WVQ-relation. The adjustment of the stream segment (i) is done with the following equations 2 to 4:

$$W'_i(t) = W_i(t) - \Delta W_{min} \tag{2}$$

$$V'_i(t) = f\left(W'_i(t)\right) \rightarrow \text{Derivation of the WVQ-relations} \tag{3}$$

$$\Delta V_i(t) = V_i(t) - V'_i(t) \tag{4}$$

where i' indicates the adjusted stage in the stream segment (i). This results in a difference of volume $\Delta V_i(t)$ which is routed to a linked spatial data structure (for example retention areas). This calculation routine is indicated with (A). Otherwise, the backwater is directly routed to the upstream linear data structure $(i-1)$. These calculation routines are indicated as (B) and (C) in Fig. 6 of the main article.

## S7. Explanation of output parameters

The results of linear data structures are given per downstream junction node. One cross section is defined per stream segment in the main flow direction from upstream to downstream point of view. The output parameters of the calculation code Kalypso-NA are written in ASCII files. These files are post-processed with the module Kalypso Hydrology to analyse the results (namely hydrographs of discharges and storage volumes) of the meso scale data structures. Additional output parameters of the implemented methods, are written in ASCII files of the defined extension folder (see Fig. S2). The output parameters are

processed for the purpose of model evaluation and application. The output parameters of flood routing computations are the formal parameters and the WVQ-relations before and after the calculation of backwater effects with the KM1- and KM5-method. A comparison of the inflow and outflow hydrograph volumes per stream segment is provided for evaluation purpose. Additionally, the time series with a temporal resolution of Δt for the following output parameters per stream segment and control structure are given:

- Storage volume before and after backwater effect computation ($m^3$).
- Water level before and after backwater effect computation (m a.s.l.).
- Control system settings per time step (-).
- Discharge per junction node ($m^3/s$).
- Evaporation rates from open water surfaces (mm/Δt).
- Exceedance flow of reservoir stream segments ($m^3/s$).

For evaluation purposes in the form of mass-conservation and for checking the calculated flood routing parameters, the following output is given per stream segment:

- Total inflow hydrograph volume ($m^3$).
- Total outflow hydrograph volume ($m^3$).
- Change in water storage per simulation run ($m^3$).
- Computed (formal) parameters of the KM1-method:
  - Retention coefficient $K_{km}$ (s).
  - Characteristic length $L_c$ (m).
  - Number of characteristic lengths n (-).
- Computed (formal) parameters of the hydraulic capacity per stream segment (in the form of WVQ-relations):
  - Water level (m).
  - Volume ($m^3$).
  - Discharge ( $m^3/s$ ).
  - Wetted cross section (m2).
  - Hydraulic radius (m).
  - Flow velocity (m/s).

### S8. Supplement results of the model evaluation for the tidal lowland catchment 'Vier- und Marschlande

Diagrams of the closure and opening state of the control elements as well as simulated and observed water levels at the gauges for two flood events: February 2002 and February 2011, are presented. The results of gauging stations are given from downstream to upstream ones. A comparison of maximal peak values is summarised in Table S1.

**Table S1: Summary of simulated and observed maximal water levels of stream segments in the Dove-Elbe streams of the 'Vier- und Marschlande catchment. Locations of gauging stations are given in 8 of the main article.**

| Gauging station | Event | Max. water level (W) in m a.s.l. | | Δ W (m) |
|---|---|---|---|---|
| | | Computed | Observed | |
| Gauge node Allermöher Deich | Feb 02 | 1.70 | 1.72 | 0.02 |
| | Feb 11 | 1.42 | 1.42 | 0.00 |
| Downstream Krapphof sluice | Feb 11 | 1.47 | 1.54 | 0.07 |
| Upstream Krapphof sluice | Feb 11 | 2.04 | 2.15 | 0.11 |
| Upstream Sehrran weir | Feb 11 | 3.29 | 3.17 | 0.12 |
| Gauge Möörkenweg | Feb 11 | 4.73 | 4.72 | 0.01 |

∅ =   0.04   m

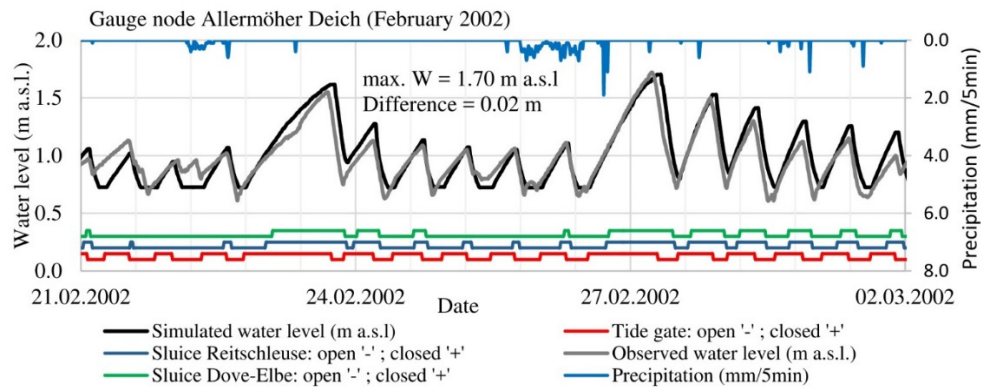



**Figure S7: Closure and opening state of the control structures as well as simulated and observed water levels at the downstream gauge 'Allermöher Deich'**

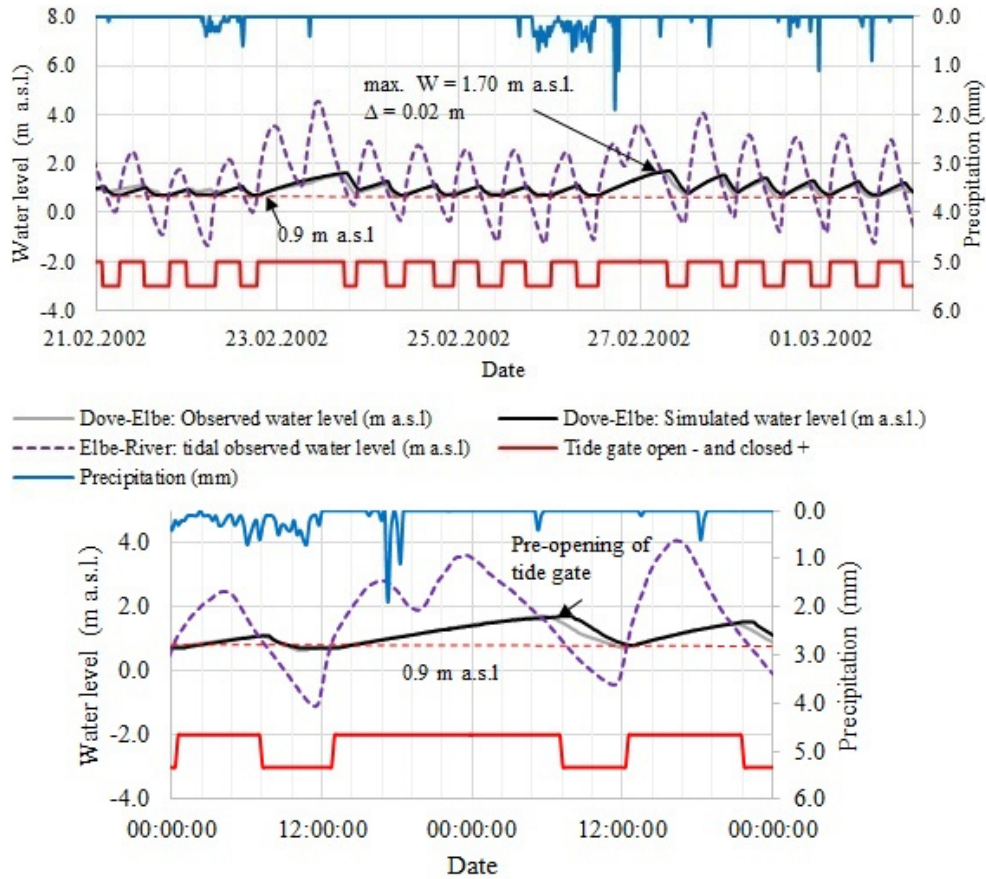



**Figure S8: Supplement detail of the flood event February 2002: Closure and opening state of the control structure as well as simulated and observed water levels at the downstream gauge 'Allermöher Deich'. The tide gate is exceptionally manually pre-opened 1.5 hours earlier, when the Elbe water level was still above 0.9 m a.s.l. Because of a difference to the automatic control structure function, the simulated results differ from the observed water levels. The computation of RMSE or the analysis with a scatter plot (R²) is not meaningful for this situation.**

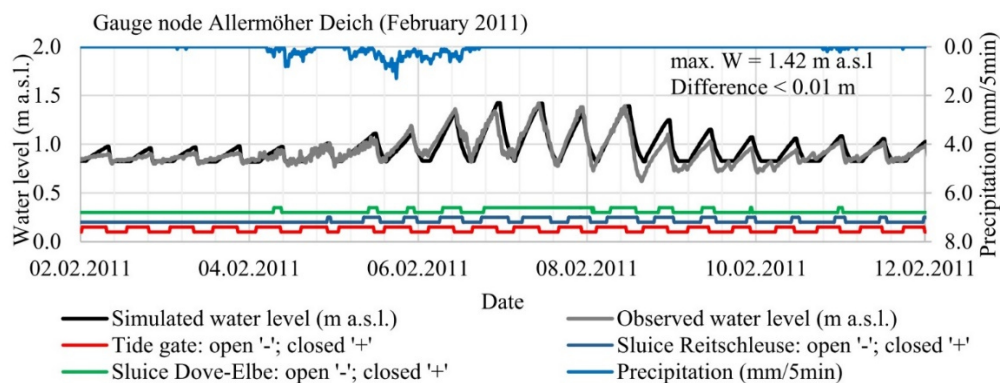



**Figure S9 Closure and opening state of the control structures as well as simulated and observed water levels at the downstream gauge 'Allermöher Deich'**

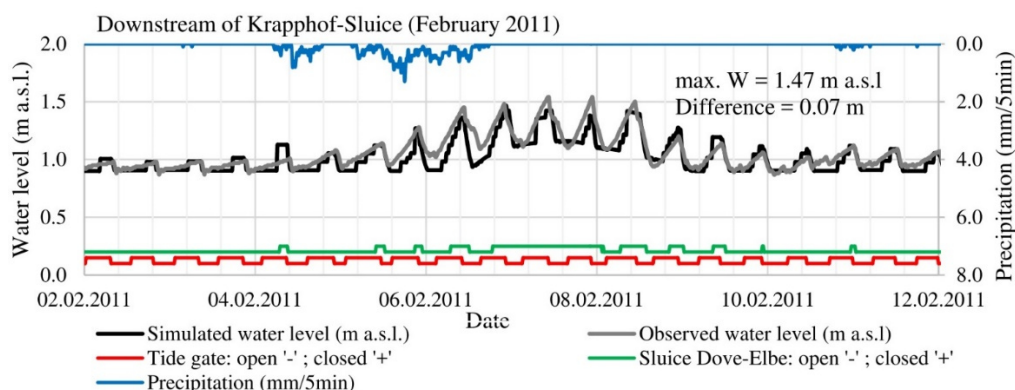



**Figure S10 Closure and opening state of the control structures as well as simulated and observed water levels at the gauge Krapphof-Sluice (Downstream).**

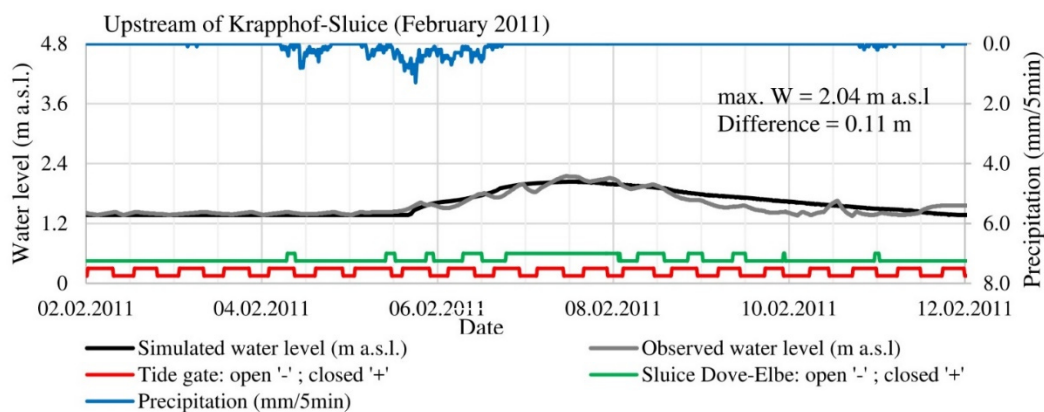



**Figure S11: Closure and opening state of the control structures as well as simulated and observed water levels at the gauge Krapphof-Sluice (Upstream).**

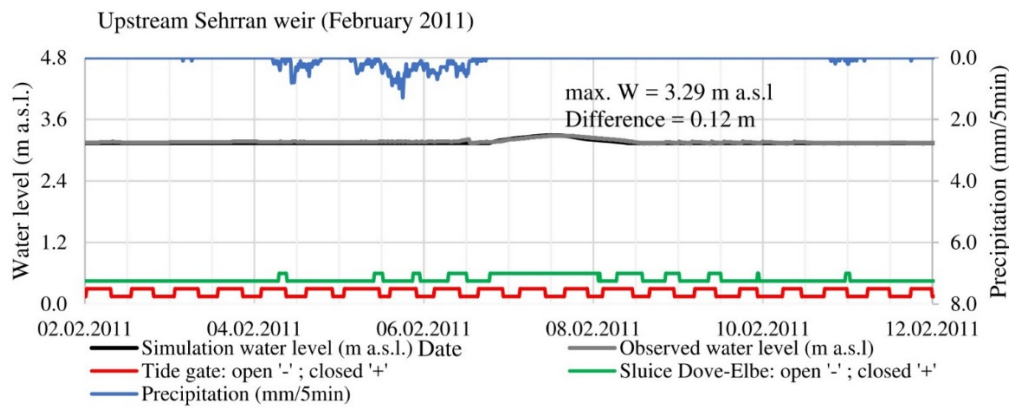



**Figure S12: Closure and opening state of the control structures as well as simulated and observed water levels at the gauge Upstream Sehrran weir.**

270

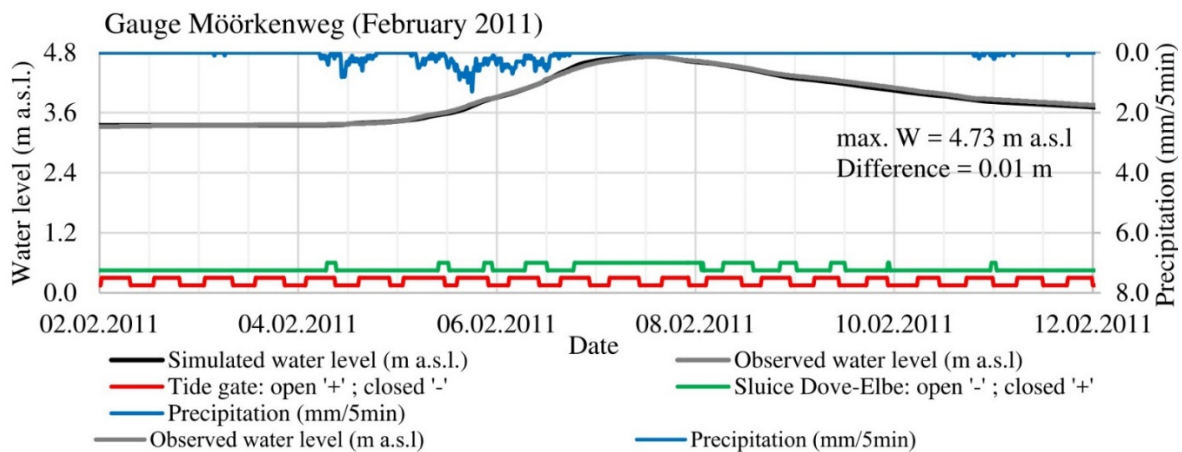
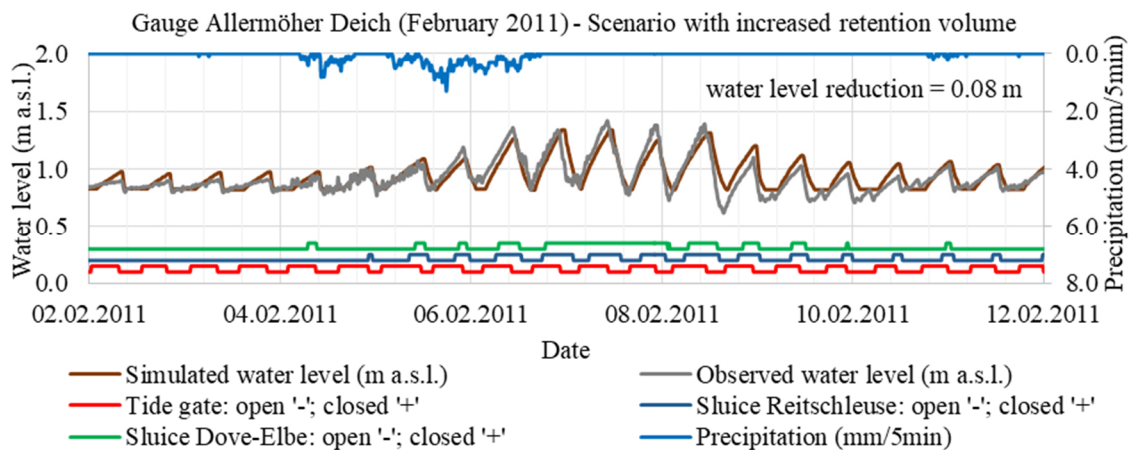



**Figure S13 Closure and opening state of the control structures as well as simulated and observed water levels at the gauge Möörkenweg.**

275 **Figure S14 Results of the scenario with increased retention volume in the forelands of the Dove-Elbe stream.**

## S9. References

Belger, G., Haase, M., Jung, T., and Lippert, K.: A GIS-based Platform for Environmental and Water resources Modeling - Kalypso Open Source, geoinformatics, 2009, 36–39, 2009.

Euler, G.: Ein hydrologisches Näherungsverfahren für die Berechnung des Wellenablaufs in Kreisrohren, Wasser und
280    Boden, 1983, 1983.

Hellmers, S., Belger, G., and Fröhle, P.: Mapping of multiple linked green infrastructure systems in rainfall-runoff models, in: Proceedings 12th International Conference on Hydroscience & Engineering: Hydro-Science and Engineering for Environmental Resilience, edited by: NCKU, Tainan, Taiwan, 2016.

Hellmers, S.: Integrating local scale drainage measures in meso scale hydrological modelling of backwater affected catchments, Dissertation, Institute of River and Coastal Engineering, Technical University Hamburg, Hamburg, Germany, 2020.

Hellmers, S. and Fröhle, P.: Integrating local scale drainage measures in meso scale catchment modelling, Water (Special Issue: Hydroinformatics and Urban Water Systems), 2017, available at: doi: 10.3390/w9020071, 2017.

Kalinin, G. P. and Miljukov, P. I.: O raschete neustanovivshegosya dvizheniya vody v otkrytykh ruslakh: On the computation of unsteady flow in open channels (in Russian), Met. i. Gidrologiya Zhuzurnal. Leningrad, Russia, 10, 10–18, 1957.

Koussis, A. D.: Assessment and review of the hydraulics of storage flood routing 70 years after the presentation of the Muskingum method, Hydrological Sciences Journal, 54, 43–61, https://doi.org/10.1623/hysj.54.1.43, 2009.

Lippert, K., Jung, T., Belger, G., Haase, M., and Schrage, N.: Kalypso: Ein Open Source Modellierungssystem für die Wasserwirtschaft, in: Wasserstandsinformationsdienste der BfG für die Bundeswasserstraßen: Kolloquium am 24. März 2009 in Koblenz, Koblenz, Germany, 2009.

Maniak, U.: Hydrologie und Wasserwirtschaft: {(in German)}, Springer Berlin Heidelberg, Berlin, Heidelberg, 2016.

Schröder, R. and Lippert, K.: Parameternachführung im Hochwasservorhersagemodell Weiße Elster, Veranstaltungen der Bundesanstalt für Gewässerkunde, 2016, 1–11, 2006.

Shreve, R. L.: Infinite topologically random channel networks, J. Geol. (Journal of Geology), 1967, 178–186, 1967.